EXPANDING UNIVERSAL PLUG AND PLAY CAPABILITIES IN POWER CONSTRAINED ENVIRONMENT

FIELD OF THE INVENTION

[0001] This invention generally relates to communication systems and more specifically to expanding universal plug and play capabilities in power constrained environment

BACKGROUND ART

[0002] Universal plug and play (UPnP™) technology defines an architecture for pervasive peer-to-peer network connectivity of intelligent appliances, wireless devices, and PCs. It is designed to bring easy-to-use, flexible, standards-based connectivity to, e.g., ad-hoc or unmanaged networks whether in the home, in a small business, public spaces, or attached to the Internet. The UPnP technology provides a distributed, open networking architecture that leverages TCP/IP and the Web technologies to enable seamless proximity networking in addition to control and data transfer among networked devices.

[0003] The UPnP Device Architecture (UDA) is designed to support zero-configuration, "invisible" networking, and automatic discovery for a breadth of device categories from a wide range of vendors. This means a device can dynamically join a network, obtain an IP address, convey its capabilities, and learn about the presence and capabilities of other devices.

[0004] A Berkeley socket application programming interface (API) comprises a library for developing applications written in C programming language that access a computer network. Berkeley sockets, also known as the BSD (Berkeley software distribution) socket API originated with the 4.2BSD system as an API. The BSD socket API forms the de facto standard abstraction for network sockets. Most other programming languages use a similar interface as the C API.

[0005] Existing standards (e.g., UPnP protocols) in the Digital Home are not friendly towards constrained devices that have a limited battery lifetime and also limited bearer bandwidth (e.g. BLUETOOTH). The UPnP protocols do not have power saving requirements during design time which prevents mobile constrained devices to use power saving techniques in order to extend a battery lifetime. The UPnP protocols allow a device to be active/present in the network, in which case the device needs to have the communication subsystem up (i.e., powered) all the time, or otherwise to be disconnected from the network. Obviously this limited behavior is not enough for constrained devices that want to preserve power by intelligently adapting the power states of communication subsystems.

[0006] Terminals can be connected to the network using a wired or a wireless (including, e.g., BLUETOOTH) technology. Some of the terminals are always connected to a power source (e.g., desktop computers, TV sets, personal video recorders, etc.), some are battery powered (e.g., mobile phones, personal digital assistants, etc.), and some can have both characteristics (e.g., laptop computers). In order to increase the lifetime of a battery for portable electronic devices (battery operated), a quite common approach is to put the communication subsystem in a low power mode state, e.g., a power saving mode for a WLAN (wireless local area network) or a sniff and hold mode for a BLUETOOTH. While the communication subsystem (e.g., a terminal) is in a power saving mode, the ability to transmit and receive messages over the network to/from that subsystem through the network is dramatically reduced, making the responsiveness of that subsystem very low or non-existent. To compensate this drawback, an access point of the network is usually caching the messages until the sleeping device is woken-up.

[0007] To further enhance this power saving scheme, interworking functions can be added to access points so that they can intelligently filter the communications cached in the buffer. One of the interworking functions that can be implemented in access points operating in the UPnP networks is to filter the UDP (user datagram protocol) multicast traffic for the devices that are in the power saving mode. Moreover, the interworking function detects when the TCP traffic is intended for the terminals which is, e.g., in a sleeping mode and can immediately start the wake-up procedure for the respective device. This prior art procedure is described, e.g., in Finnish patent application No. 20040179 "Optimization of Data Transfer between Networked Devices", by Vlad Stirbu and Mika Saaranen, filed on Feb. 6, 2004

[0008] Furthermore, the sleeping mode (or similarly an intermediate mode, i.e., less active than normal but more active than the sleeping mode) provides high energy saving level with assumption that the terminal (such as a mobile device) can be woken-up in a reasonable time. Even in case of a standby mode exploiting bearer level energy savings, where the link can be activated by the incoming traffic, it requires unreasonably significant time to wake up. Sleeping mode is by its definition slower (slower than, e.g., less stringent modes), i.e., it requires more time to wake up the device and therefore more TCP protocol delays and re-transmissions occur. So, because the receiving device is in the standby mode, it takes longer time for it to respond. This time may (or generally will) exceed a TCP retransmission time or even a socket time out may occur. The socket time out is a part of the socket implementation defining the time that can be used for establishing a connection before giving up this attempt. This is separate time out mechanism than defined for TCP protocol itself. This leads to a poor usage of networking resources, quite frequently causing spurious errors visible to the users. This is, naturally, leading to a degraded user experience.

DISCLOSURE OF THE INVENTION

[0009] The object of the present invention is to provide a method for expanding universal plug and play capabilities in a power constrained environment thus facilitating communication between two terminals with available universal plug and play application capabilities on the terminals, wherein one of them is in a sleeping mode.

[0010] According to a first aspect of the invention, a method for communicating of a terminal with a further terminal through a network in a power constrained environment using a communication application available in the terminal, comprises the steps of: receiving by the terminal a notification about a power saving mode of the further terminal; providing to the network by a protocol stack of the terminal a communication signal intended for the further terminal by transmitting the communication signal according to a pre-selected procedure; and receiving by the terminal, after the further terminal is woken up, an acknowledgement of receiving the communication signal by the further terminal, wherein the terminal waits for the acknowledgement at least for a time interval needed for that further terminal to wake up before